CTU：加计数

说明

可以使用“加计数”指令，递增输出 CV 的值。 如果输入 CU 的信号状态从“0”变为“1”（信号上升沿），则执行该指令，同时输出 CV 的当前计数器值加 1。 每检测到一个信号上升沿，计数器值就会递增，直到达到输出 CV 中所指定数据类型的上限。 达到上限时，输入 CU 的信号状态将不再影响该指令。

可以查询 Q 输出中的计数器状态。 输出 Q 的信号状态由参数 PV 决定。 如果当前计数器值大于或等于参数 PV 的值，则将输出 Q 的信号状态置位为“1”。 在其它任何情况下，输出 Q 的信号状态均为“0”。

输入 R 的信号状态变为“1”时，输出 CV 的值被复位为“0”。只要输入 R 的信号状态仍为“1”，输入 CU 的信号状态就不会影响该指令。

说明
只需在程序中的某一位置处使用计数器，即可避免计数错误的风险。

每次调用“加计数”指令，都会为其分配一个 IEC 计数器用于存储指令数据。 IEC 计数器是一种具有以下某种数据类型的结构：

对于 S7-1200 CPU

系统数据类型 IEC_<Counter> 的数据块（共享 DB）	局部变量
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

对于 S7-1500 CPU

系统数据类型 IEC_<Counter> 的数据块（共享 DB）	局部变量
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • CTU_LINT / CTU_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

可以按如下方式声明 IEC 计数器：

- 系统数据类型 IEC_<Counter> 的数据块声明（例如，“MyIEC_COUNTER”）
- 声明为块中“Static”部分的 CTU_<Data type> 或 IEC_<Counter> 类型的局部变量（例如 #MyIEC_COUNTER）

在程序中插入该指令时，将打开“调用选项”(Call options) 对话框，可以指定 IEC 计数器将存储在自身数据块中（单背景）还是作为局部变量存储在块接口中（多重背景）。 如果用户创建一个单独的数据块，那么该数据块将保存到项目树“程序块 > 系统块”(Program blocks > System blocks) 路径中的“程序资源”(Program resources) 文件夹内。 有关本主题的更多信息，请参见“另请参见”。

如果在单独的数据块中设置 IEC 计数器（单背景），则将默认使用“优化的块访问”(optimized block access) 创建背景数据块，并将各个变量定义为具有保持性。 有关在背景数据块中设置保持性的更多信息，请参见“另请参见”。

如果在函数块中使用“优化的块访问”(optimized block access) 设置 IEC 计数器作为本地变量（多重背景），则其在块接口中定义为具有保持性。

执行“加计数”指令之前，需要事先预设一个逻辑运算。 该运算可以放置在程序段的中间或者末尾。

参数

下表列出了“加计数”指令的参数：

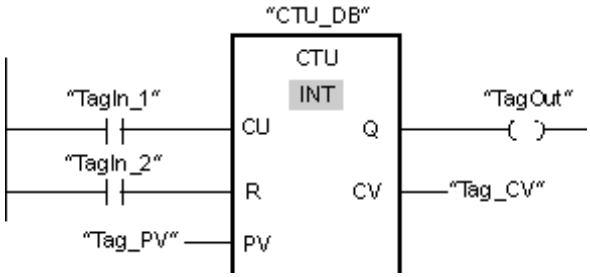
参数	声明	数据类型	存储区		说明
			S7-1200	S7-1500	
CU	Input	BOOL	I、Q、M、D、L 或常数	I、Q、M、D、L 或常数	计数输入
R	Input	BOOL	I、Q、M、D、L、P 或常数	I、Q、M、T、C、D、L、P 或常数	复位输入
PV	Input	整数	I、Q、M、D、L、P 或常数	I、Q、M、D、L、P 或常数	置位输出 Q 的值。
Q	Output	BOOL	I、Q、M、D、L	I、Q、M、D、L	计数器状态
CV	Output	整数、CHAR、WCHAR、DATE	I、Q、M、D、L、P	I、Q、M、D、L、P	当前计数器值

可以从指令框的“???”下拉列表中选择该指令的数据类型。

有关有效数据类型的更多信息，请参见“另请参见”。

示例

以下示例说明了该指令的工作原理：



当“**TagIn_1**”操作数的信号状态从“**0**”变为“**1**”时，将执行“加计数”指令，同时“**Tag_CV**”操作数的当前计数器值加 1。 每检测到一个额外的信号上升沿，计数器值都会递增，直至达到该数据类型的上限 **(INT = 32767)**。

PV 参数的值作为确定“**TagOut**”输出的限制。 只要当前计数器值大于或等于操作数“**Tag_PV**”的值，输出“**TagOut**”的信号状态就为“**1**”。 在其它任何情况下，输出“**TagOut**”的信号状态均为“**0**”。